

## Extendible Hash Index & B+ Tree

Profesor Heider Sanchez

### P1. Extendible Hash:

Dado el siguiente conjunto de claves de registros a insertar en un archivo gestionado por un hash extensible, ¿Cómo quedaría organizado los datos al final de todas las inserciones?

#### Keys

2,3,6

7,11,16

18,20,23

28,29,30

Considere como profundidad global de dos bits ( $d = 2$ ) y un factor de bloque de 4 registros ( $fb = 4$ ). Ilustre paso a paso el proceso de splitting. Sea ordenado y claro en su solución.

#### Paso 1:

00	
01	
10	
11	

0



1

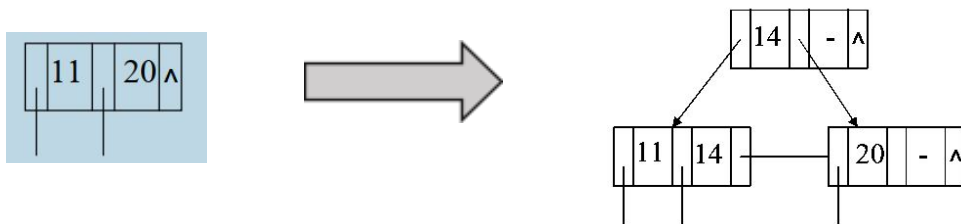


Luego diseñe el algoritmo de inserción en memoria secundaria.

## P2. B+ Tree: Inserción

Las inserciones en el árbol B+ deben garantizar la propiedad de balanceo, por lo tanto, los nodos deben tener al menos  $fb/2$  entradas ( $fb$  es el máximo número de entradas en un nodo). En el ejercicio siguiente realice las operaciones de inserción aplicando división cuando un nodo esta lleno. Considerar  $fb = 3$ ,  $fb_{leaf} = 2$ .

### Insertar 20 y 11



### Insertar 25

### Insertar 12

## P4. B+ Tree: Búsqueda

Proponga un algoritmo de búsqueda para B+ Tree cuando el índice es agrupado.

